Amendments to the Figures

The attached sheet of figures includes changes to Figure 1. Duplicate element labeling has been corrected. Marked changes to Figure 1 and a full set of formal figures are attached.

Attachment: Annotated Sheet Showing Changes

Formal Figures

Replacement Sheet

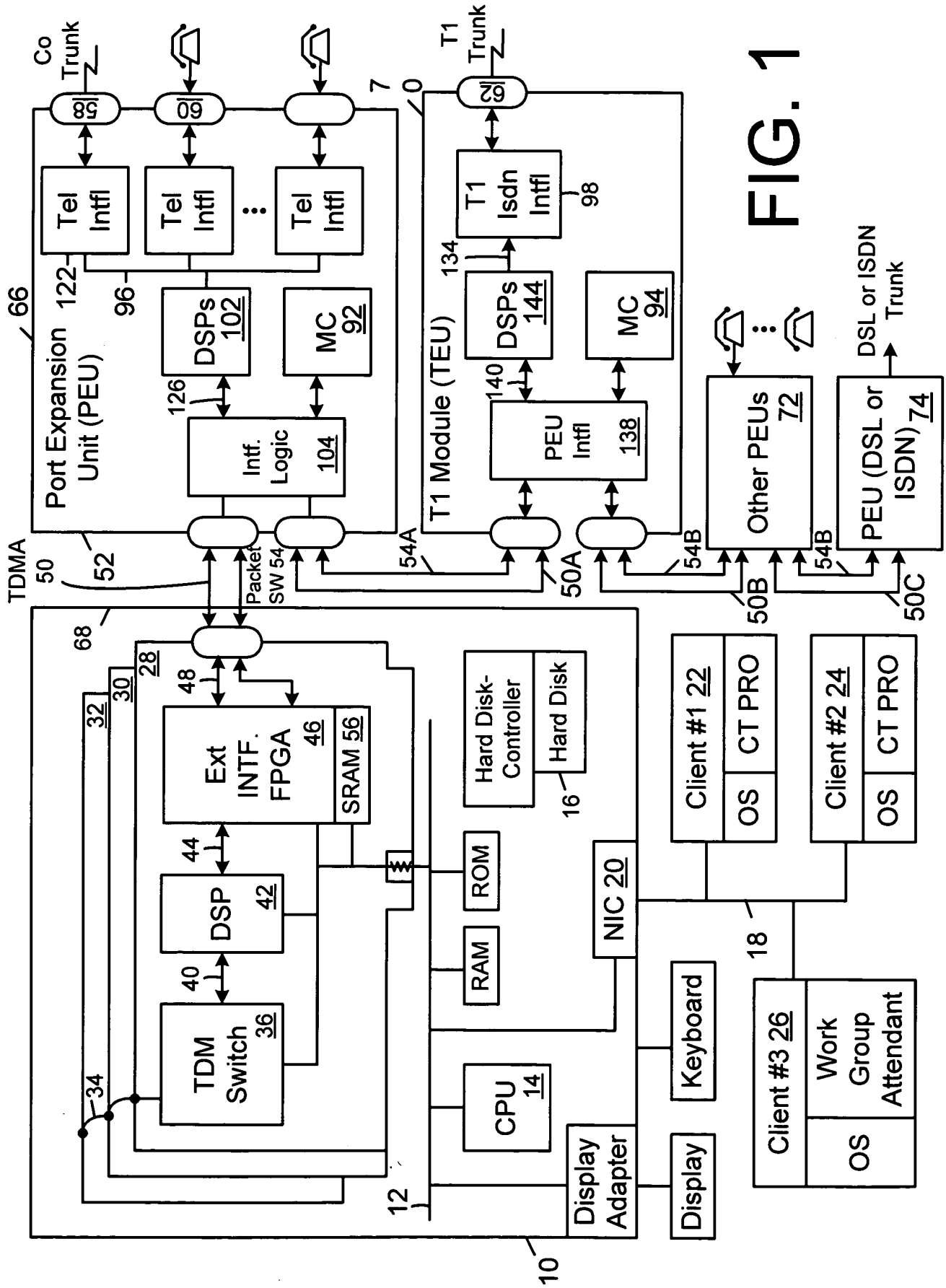


FIG. 1

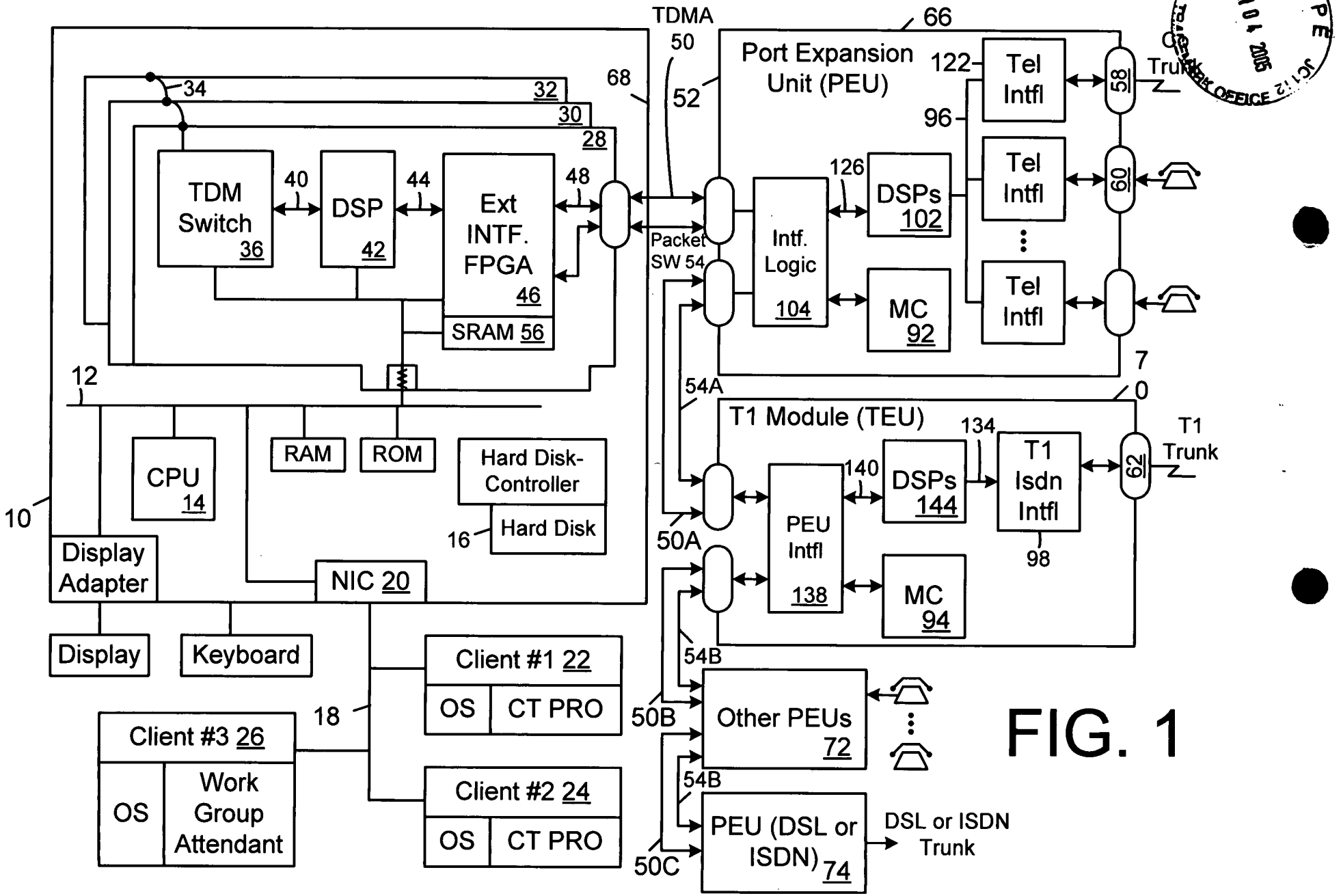


FIG. 1

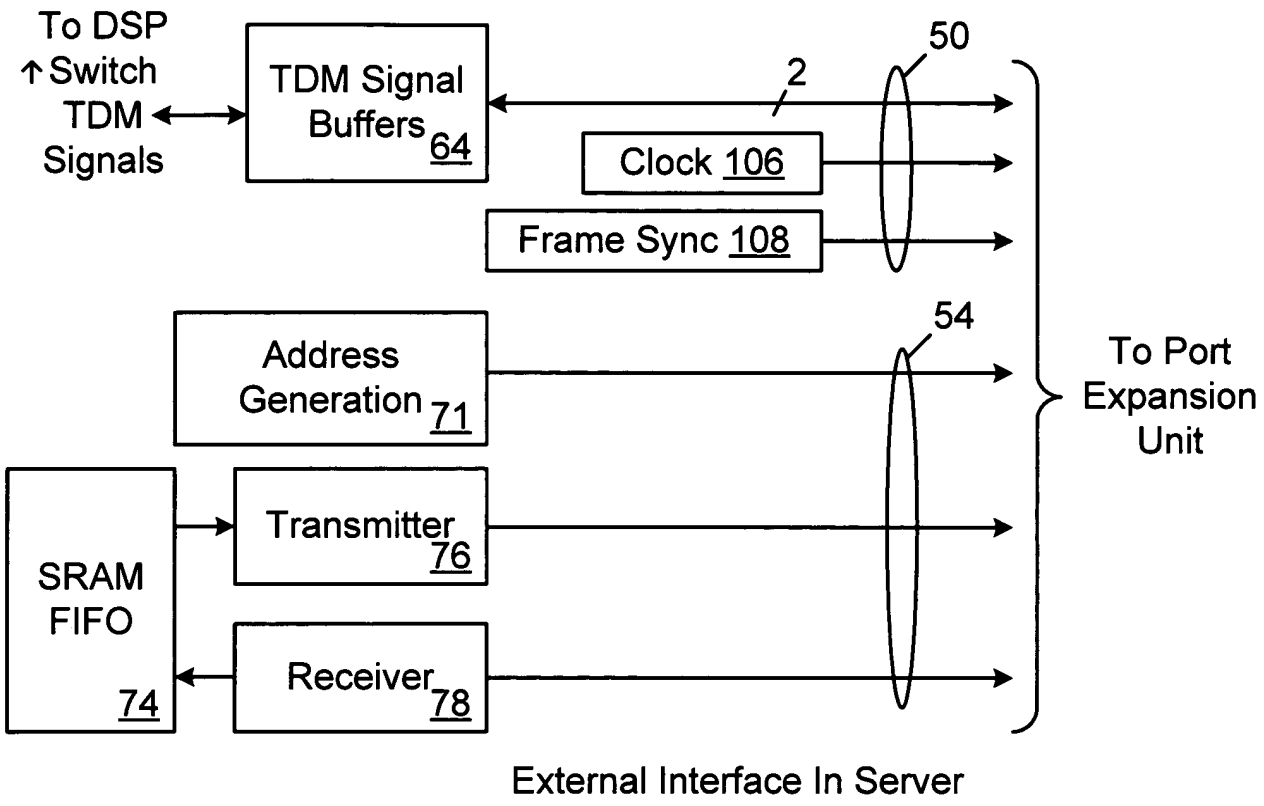


FIG. 2

SIZE 80	DA 82	SA 84	TYPE 86	PAYLOAD 88	CRC 90

FIG. 3

START BIT	4-BIT ADDRESS	MATCH BIT	STOP

FIG. 4

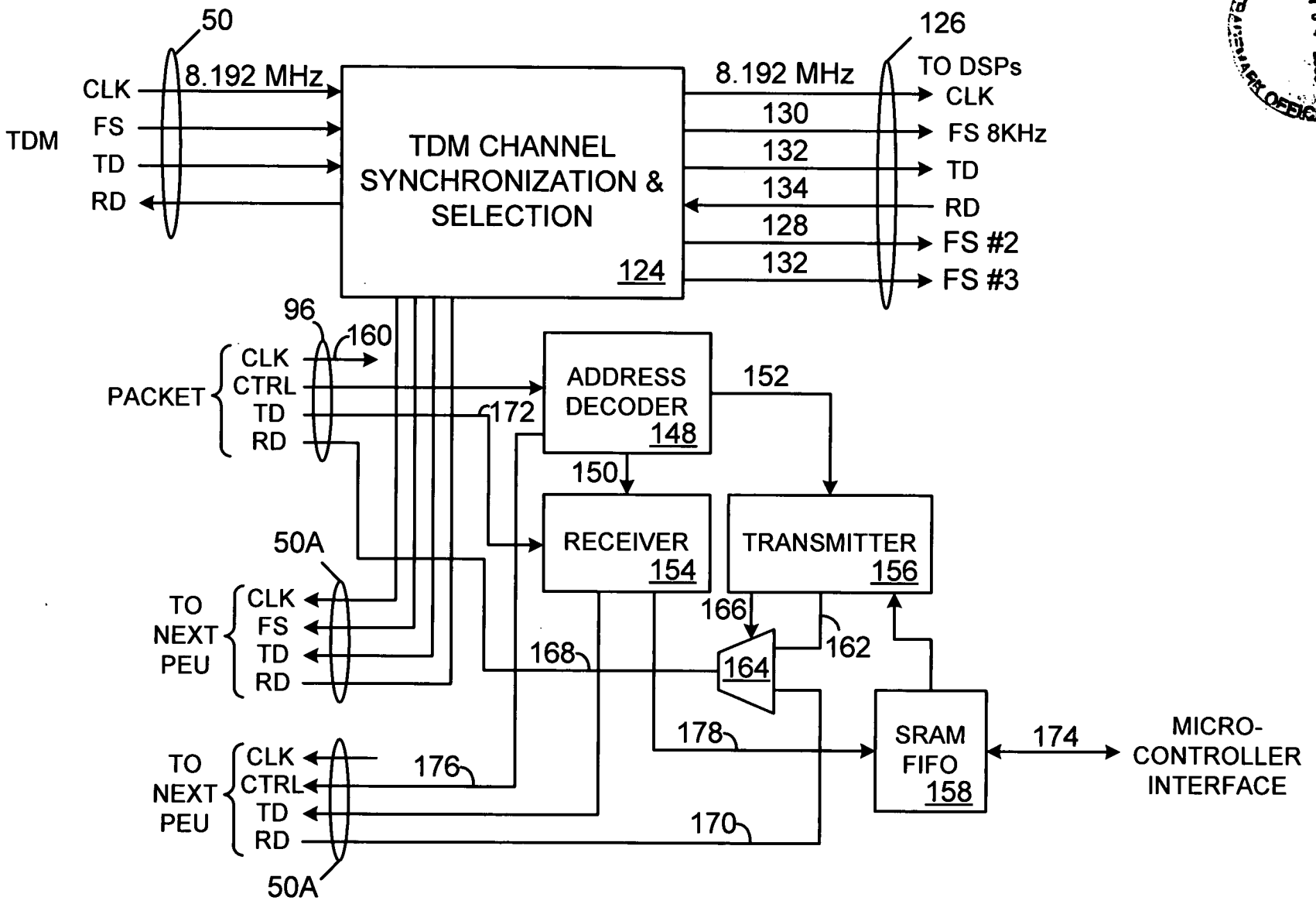
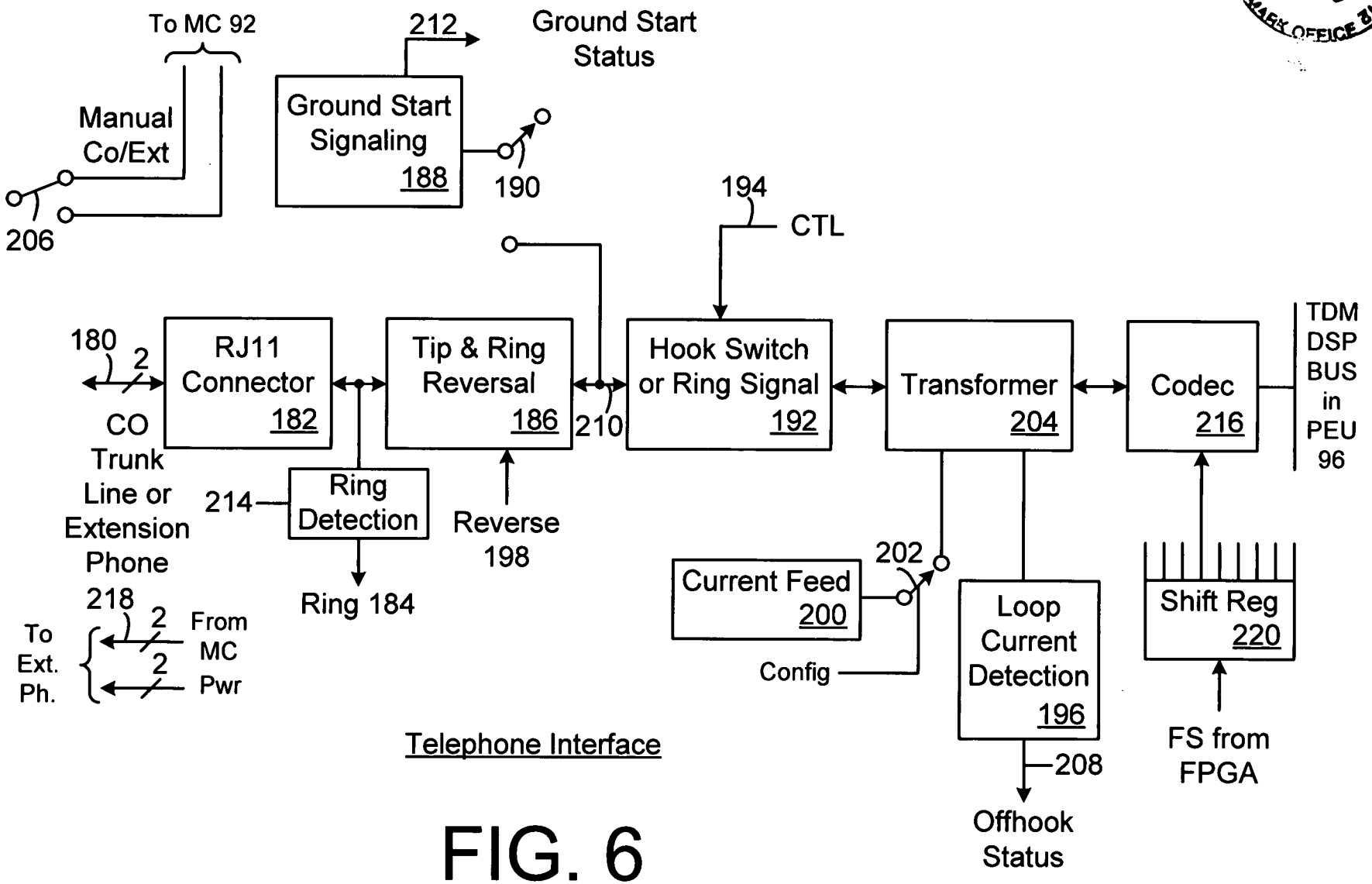


FIG. 5



Telephone Interface

FIG. 6

PEU Processing

Microcontroller (hereafter MC) loads code for DSPs and programs the FPGA 222

DSP initializes each function and waits for channel assignment from MC 224

FPGA senses clock on bus 50 and learns its address by watching the match bit of the token 226

Once the address is learned, the FPGA interrupts the MC which records the address and sets a flag that it is now operational. MC sends a packet to PBX process on host that these PEU is now operational 228

MC starts generating a predefined hello packet that is sent to PBX indicating how many ports it has

The PBX process receives the hello packet, assigns a channel on bus 50 to each port of the PEU and sends a packet back with that assignment 232

MC receives assignments and parses out assigned channels among all the DSPs on the PEU and gives the DSP a start processing signal 234

DSP buffers all 128 timeslots of data from bus 126 headed out to ports and buffers all 32 timeslots of data coming in from the ports 236

DSP also receives and buffers any tone data received from the MC or the port in a tone buffer for each port and receives the payloads of voice data from prerecorded messages and voicemail packets sent over the packet bus or received from the port in a voice buffer for each port 238

Read the timeslot data, voice data and tone data from each buffer of each port for each direction and weight data according to volume control weighting function for each type of data and sum the results 240

FIG. 7A

FIG. 7B

FIG. 7A

Write the resulting volume control weighted mixed data to the appropriate buffer 242

DTMF tone generation: MC receives packet form PBX host telling it which DTMF tones to generate on each channel 244

MC retrieves PCM data for desired tones and writes it into DSP tone buffer for the appropriate channel 246

DSP takes tone data and does volume control weighting on it and writes resulting sum data into output buffer for the channel 248

Play prerecorded announcements and voicemail data: receive control packet specifying a particular stream that is to be played on a particular port 250

PBX software broadcasts packets containing voice data to all peus over packet bus, each packet of a particular voicemail message or prerecorded message having a stream identifier with one packet being transmitted every 26 msec 252

FPGA at peu and copies any packets with correct PEU address or a broadcast PEU address that indicates all ports are supposed to process the message into the receive FIFO or MC 254

MC retrieves packets out of FIFO and looks at type field to determine if it is play data. If so, then the MC looks at the port address. If the port address is a broadcast address, then the MC looks at the stream ID and compares to stream ID previously identified in control messages for various ports to determine if there are any matches. MC writes voice data of payload section into input buffer of DSP that corresponds to any matches of stream IDs 256

FIG. 7C

FIG. 7B

FIG. 7B

If the port address is not a broadcast address, then the MC writes the payload data from the packet into the input buffer of the DSP that corresponds to the port identified in the port address 258

DSP does volume control weighting and mixing of voice data with TDM data and tone data and writes summed data to output buffers of all ports on which message is to be played 260

DSP record data: data coming in from port is stored in TDM input buffer for each port. Data going out to port is stored in TDM output buffer. Volume control mix process reads TDM data traveling in each direction for port as well as packet data and tone data and weights each set of data in accord with weighting functions established by PBX software via control packets based upon user functionality requests 262

The summed volume control adjusted data is written into the record buffer of the DSP. Once 208 bytes have been accumulated, the DSP interrupts the MC. The MC reads the record data and packetizes it and stores it into transmit FIFO of FPGA. FPGA transmits packet over packet bus 264

DSP decodes caller ID data: MC detects first ring on any co port by polling the ring detector for all the ports and sets a status in the DSP to start transferring caller ID data. The DSP is constantly looking for caller ID data 266

DSP decodes incoming caller ID data, and when one byte has been received, interrupts the MC. The MC retrieves the caller ID data and verifies the checksum. Then the caller ID data is packetized and transmitted to the PBX software process. 268

The PBX process updates the attribute data of an object in memory created for this particular call with the caller ID data 270

FIG. 7D

FIG. 7C

FIG. 7C

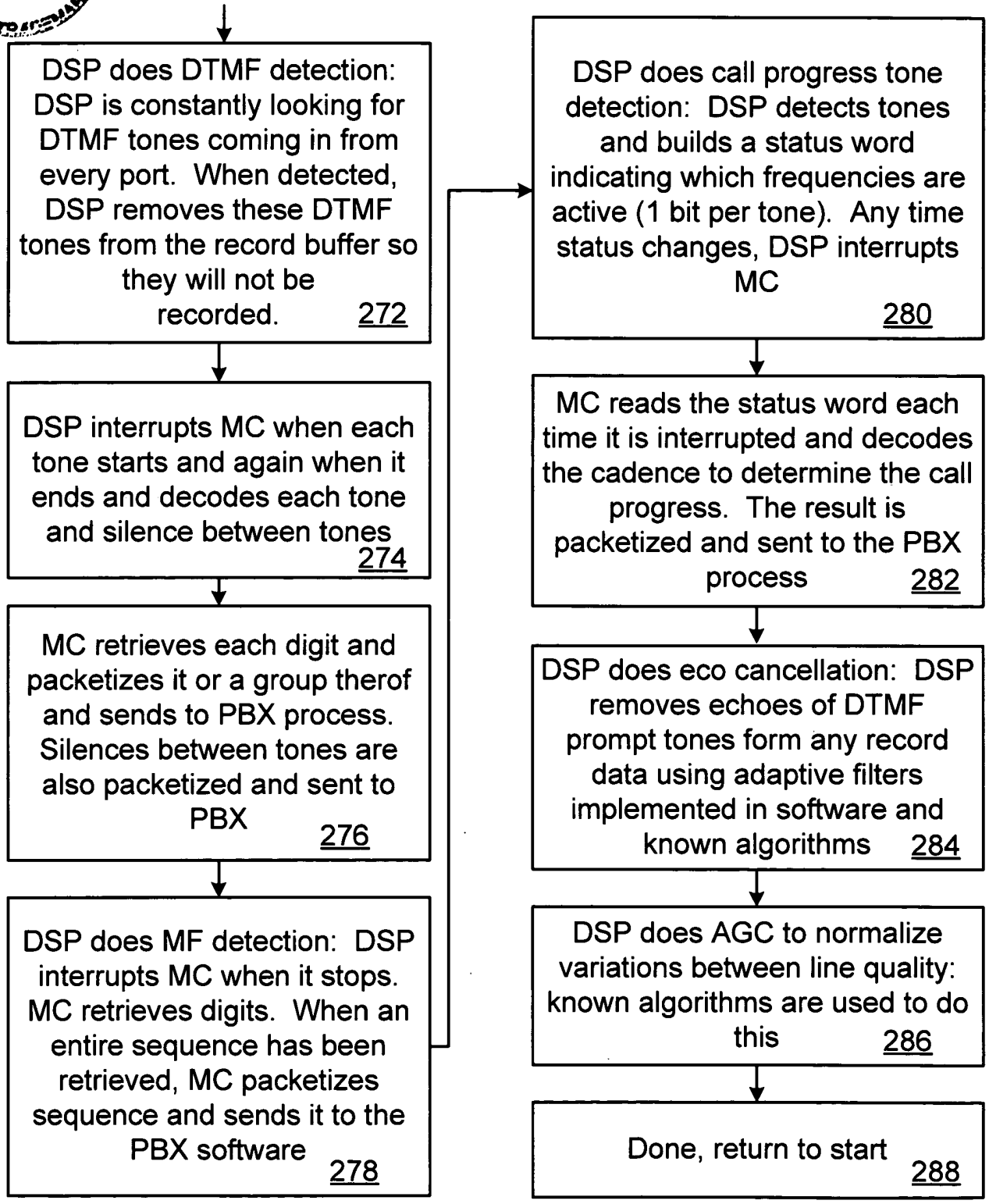


FIG. 7D

MC - Telephone Interface Processing for Incoming Call

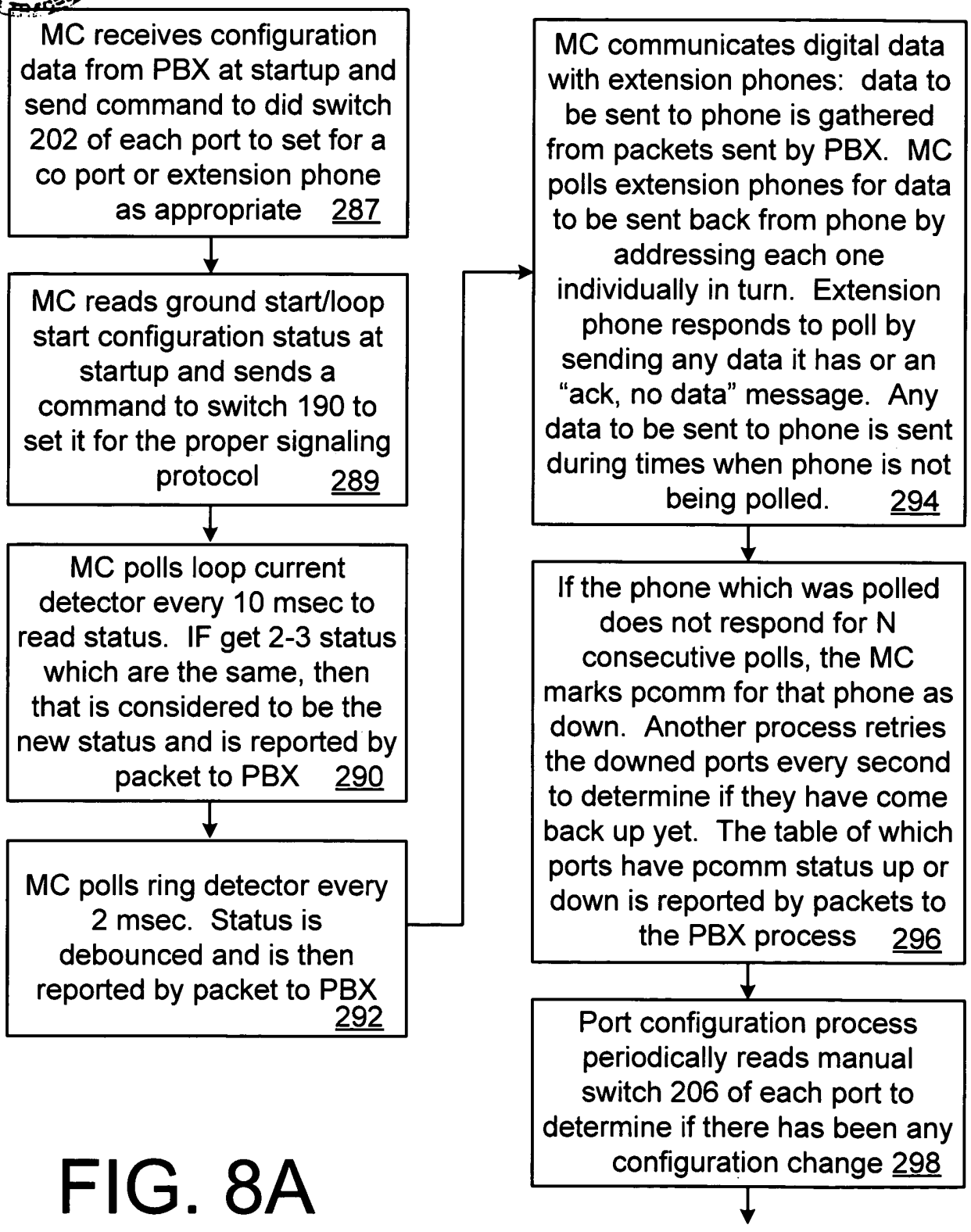


FIG. 8A

FIG. 8B

FIG. 8A

If any configuration change has happened, a packet is sent to the PBX process signaling the changed configuration 300

Monitor for incoming did call on all co ports configured for did: MC examines status of loop current detection and ring status and draws conclusion as to whether a did incoming call is ready to be sent by co 302

When the MC detects loop current and no ring signal, it sends a command to the tip and ring reversal circuit to cause it to flip the tip line voltage polarity to +48 volts relative to ring to signal the co that the PBX was ready to receive the dialed digits. The MC then sends a control packet to the PBX indicating a did call is inbound 304

Co sends dialed digits as DTMF tones. DSP detects DTMF tones and interrupts MC. MC reads digits and packetizes and sends to PBX process 306

FIG. 8B

The PBX process reads the dialed digits of the did call and looks up the extension that is mapped to that dialed number and sends a command packet to the appropriate PEU saying "ring port x" 308

MC activates line 194 to ring signal circuit 192 to cause it to start sending ring signal to extension named in control packet at cadence given in control packet 310

When the polling of the loop current detector indicates that the extension phone has gone offhook, MC sends a control packet with that change in status to the PBX process 312

PBX sends control packet back to MC saying answer call and sets up the switch connection between the co port and the extension port 314

MC responds to control packet by deactivating reverse command on line 198 to cause tip and ring reversal circuit to set tip and ring polarities back to normal to signal co that extension phone has answered call 316

FIG. 8C

FIG. 8B

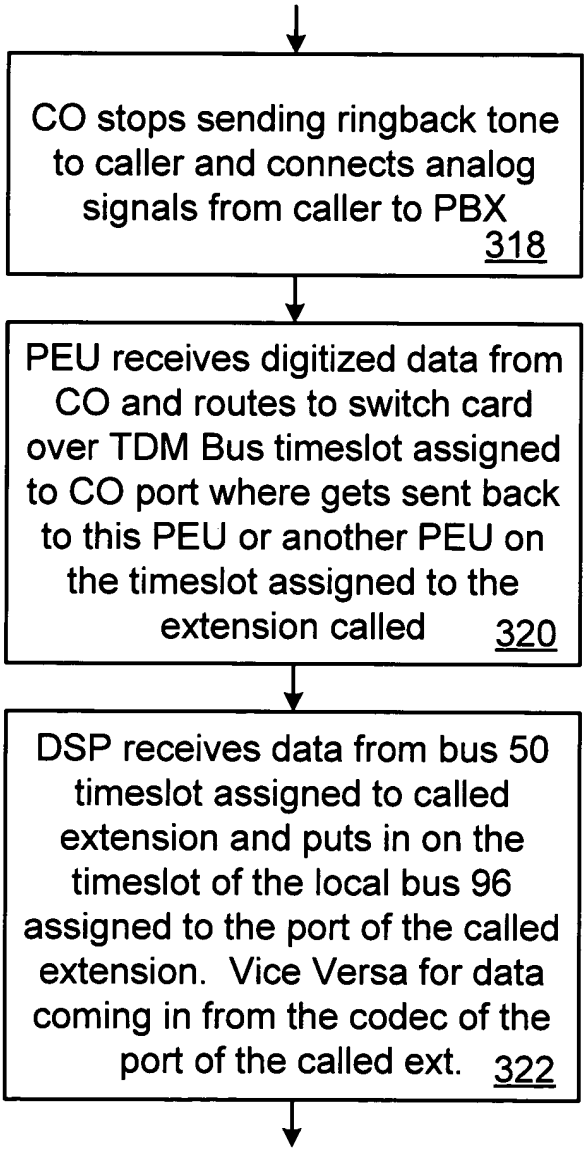


FIG. 9A

FIG. 8C

MC - Telephone Interface Processing for Incoming Call

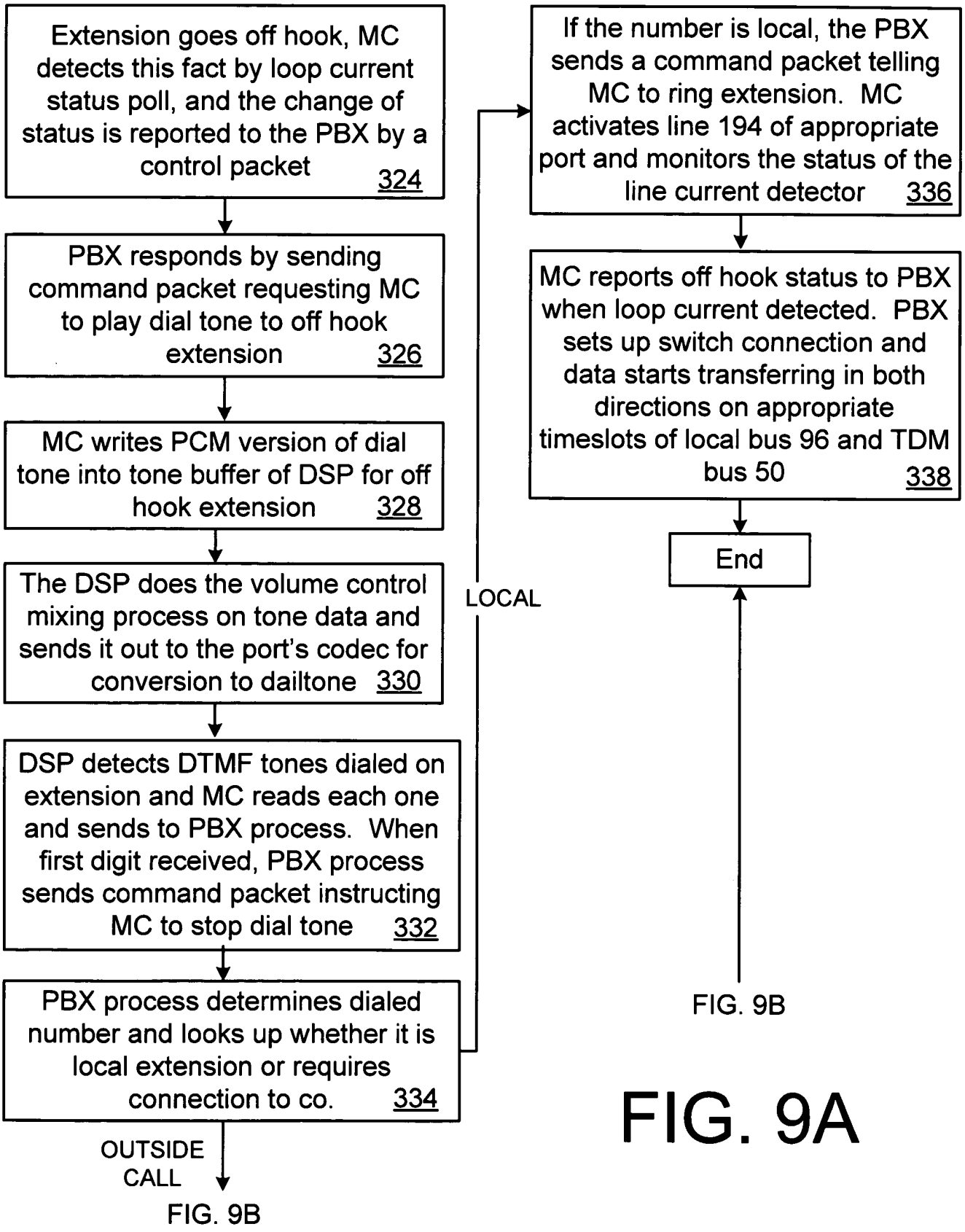


FIG. 9B

FIG. 9A

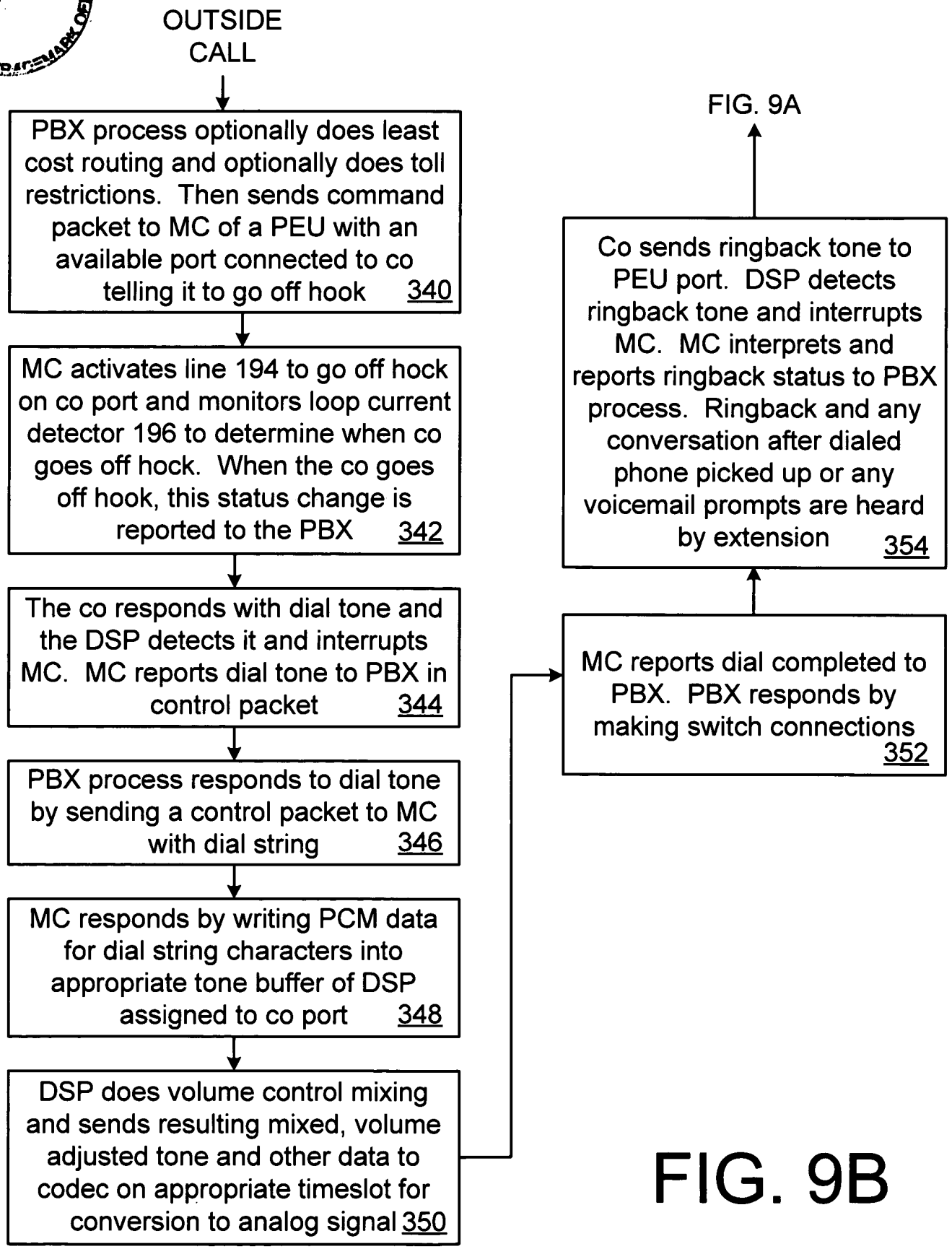


FIG. 9A

FIG. 9B

TEU Processing for an Incoming T1 Call

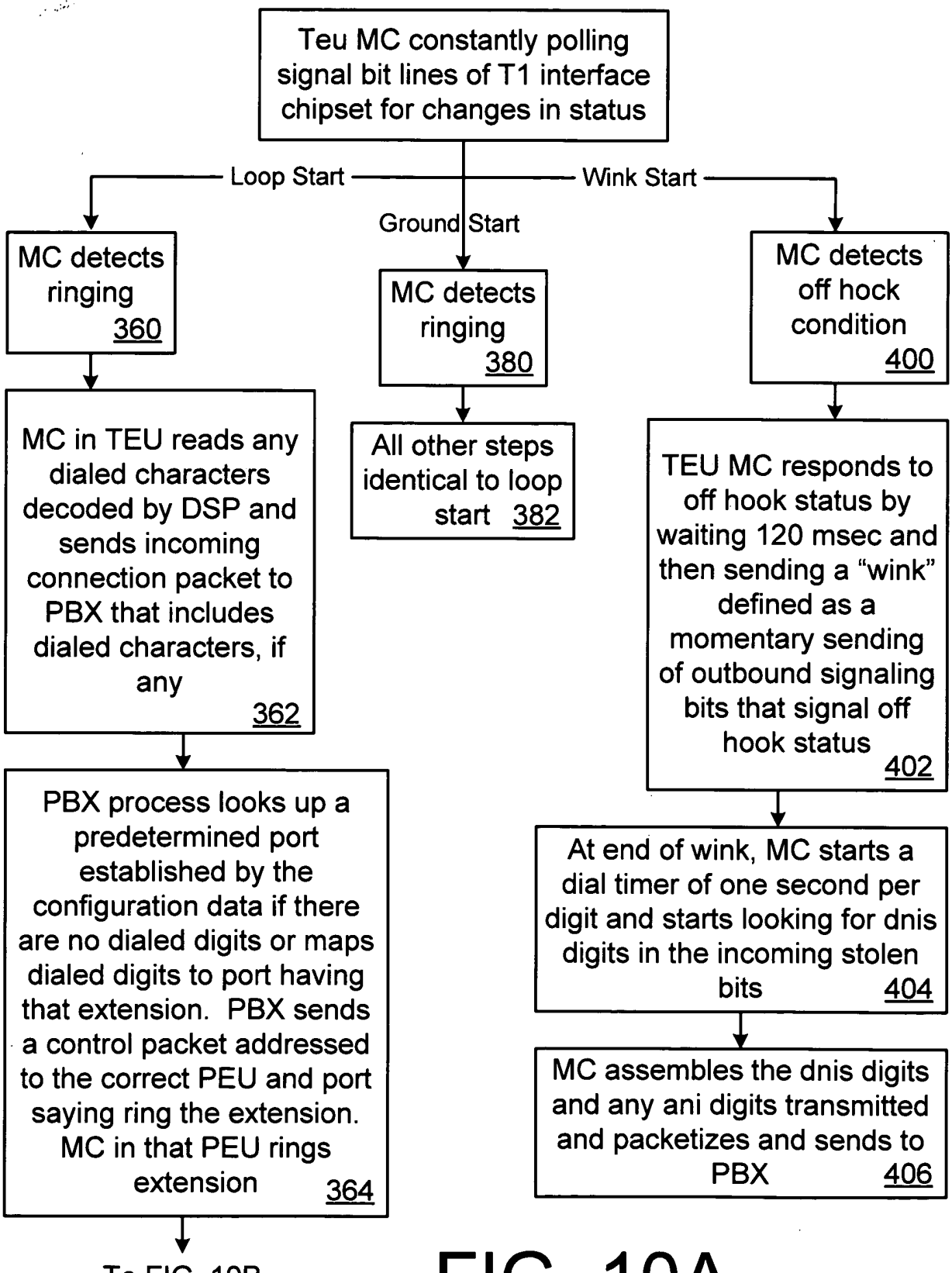


FIG. 10A

From FIG. 10A

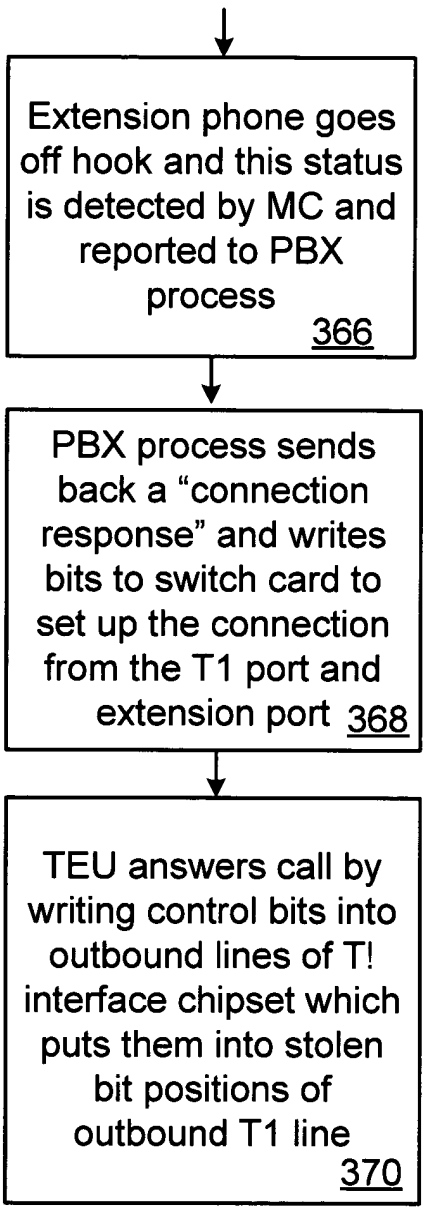


FIG. 10B

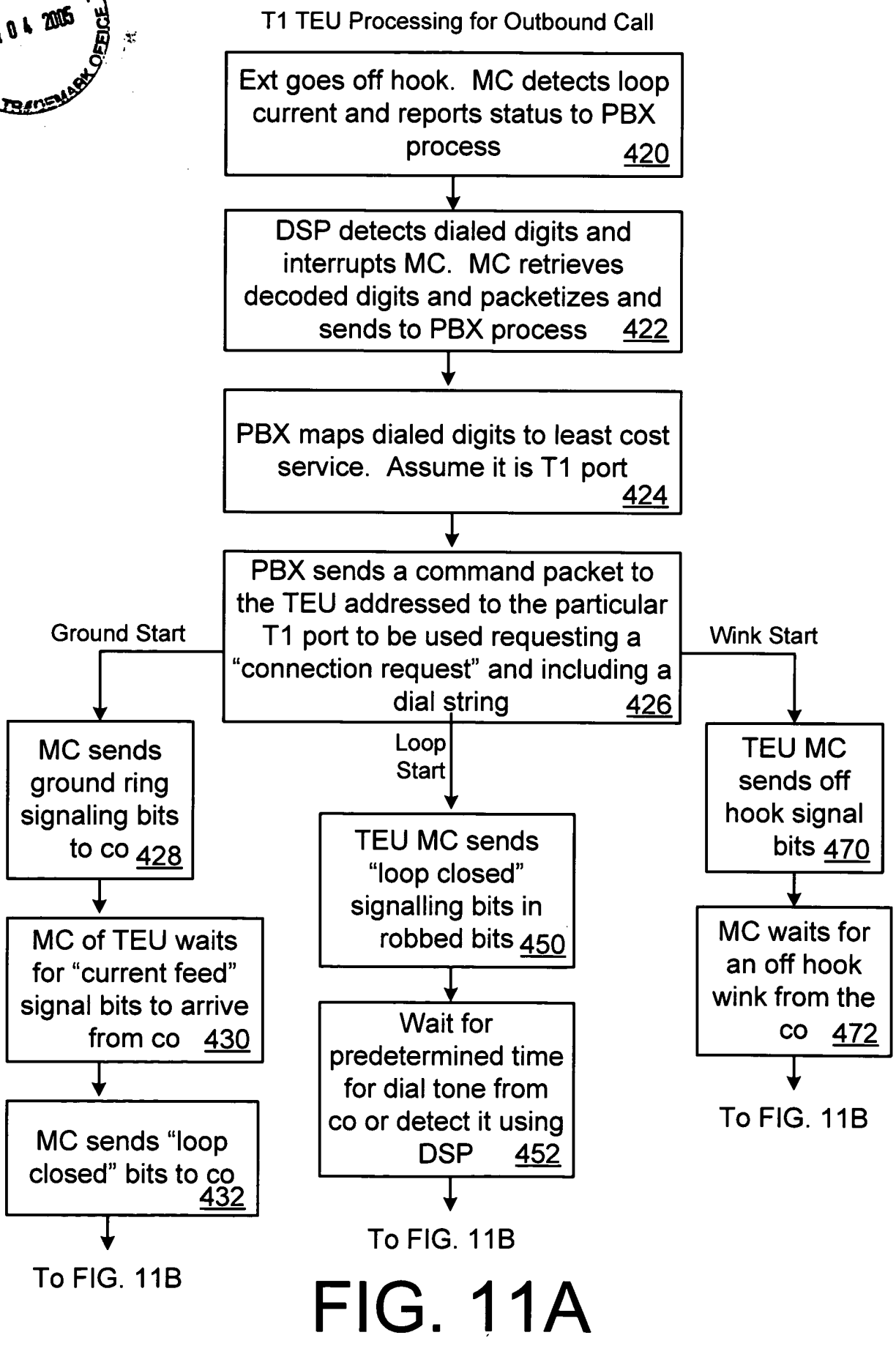


FIG. 11A

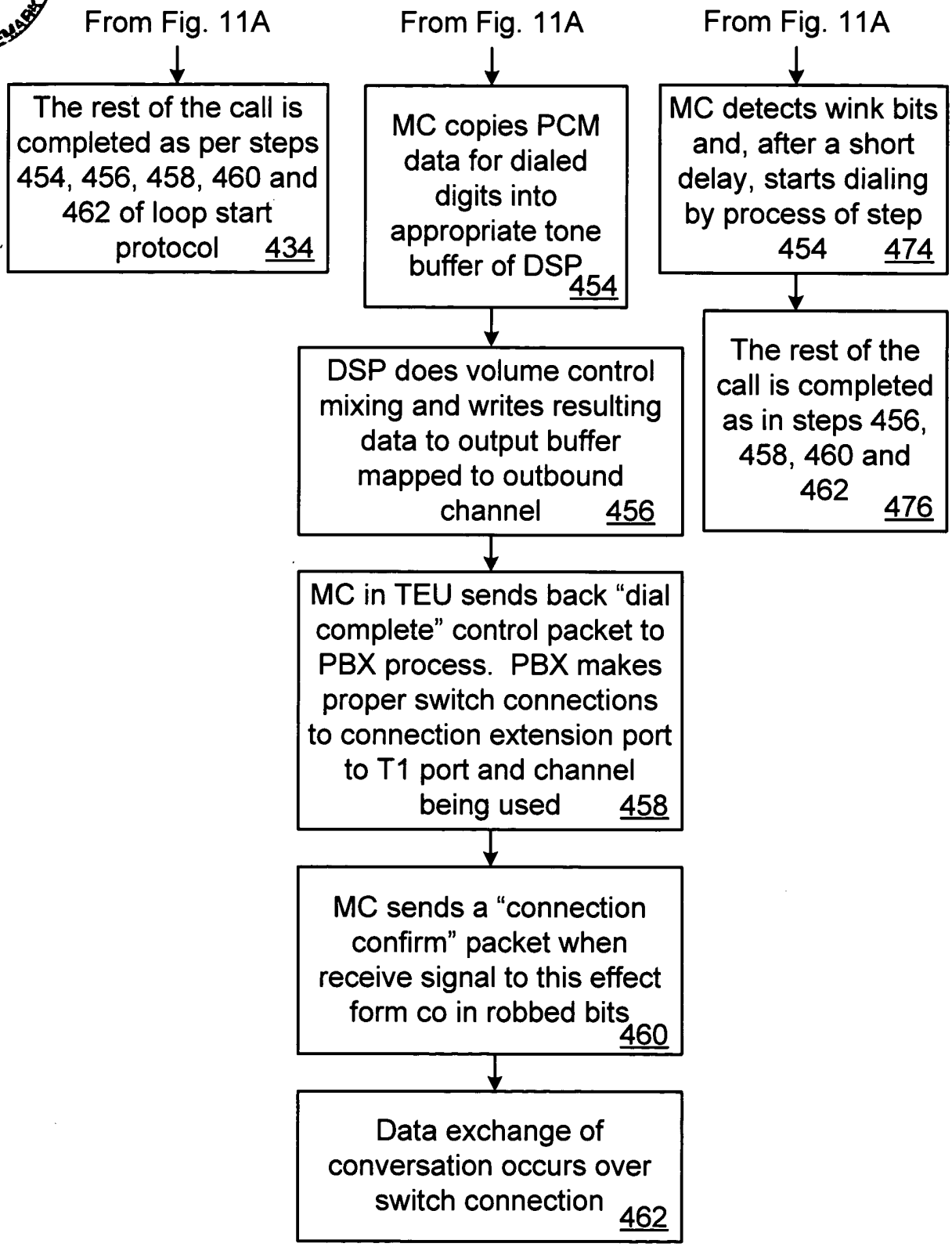


FIG. 11B

Switch Card DSP Conferencing Process

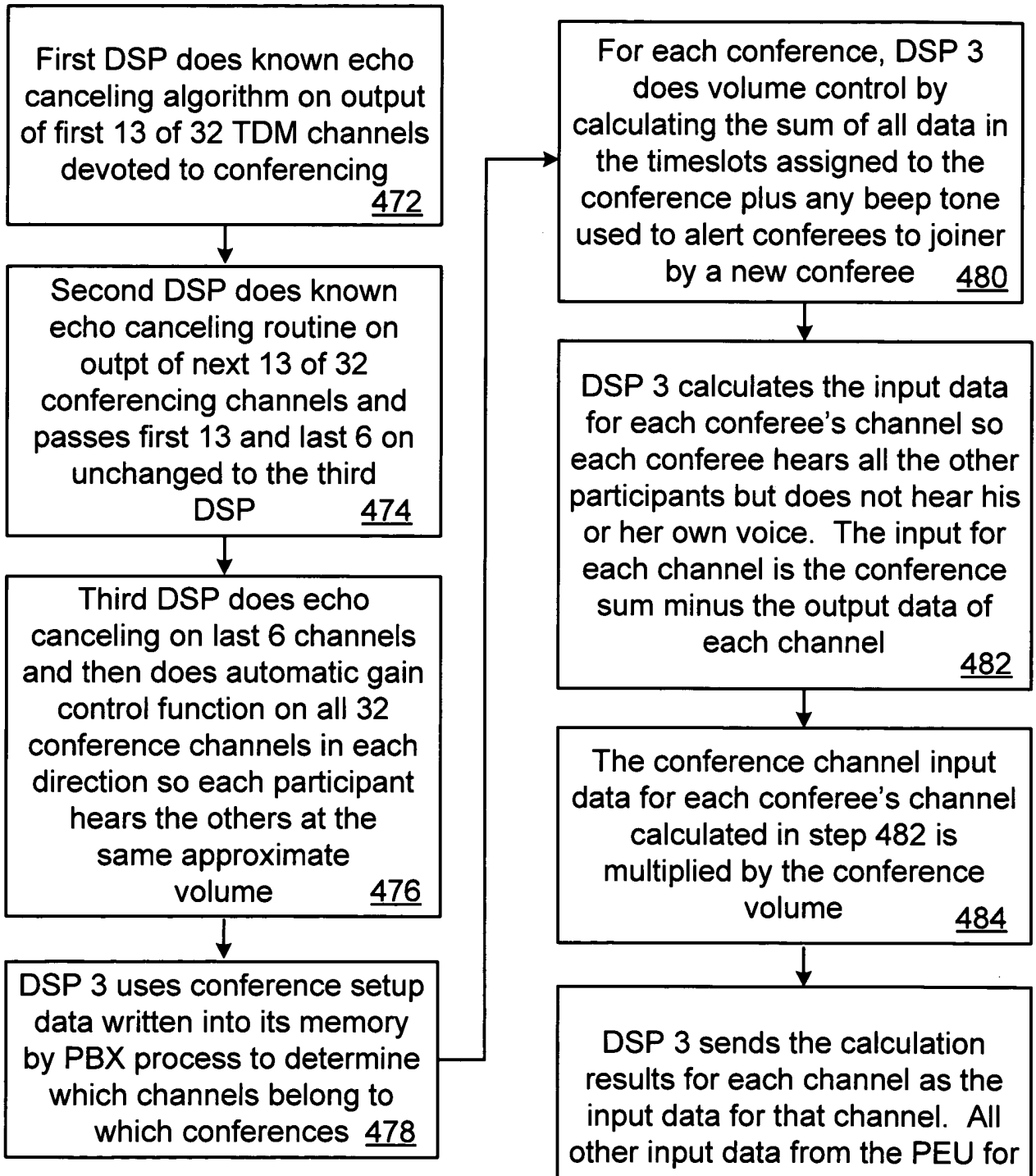


FIG. 12

Analog Loop Start Protocol Process - CO Originates Call

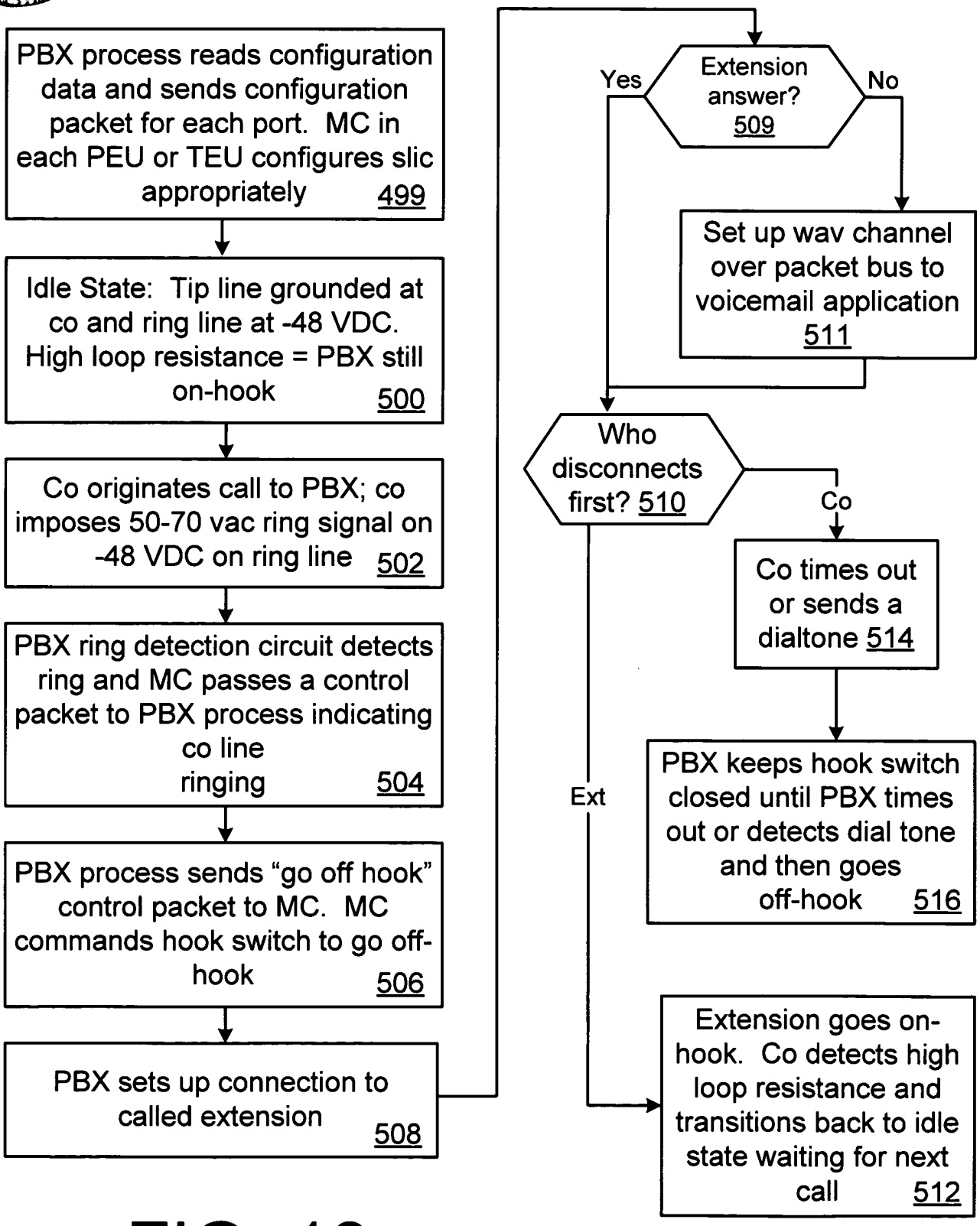


FIG. 13

Analog Loop Start Protocol Process - PBX Initiates the Call

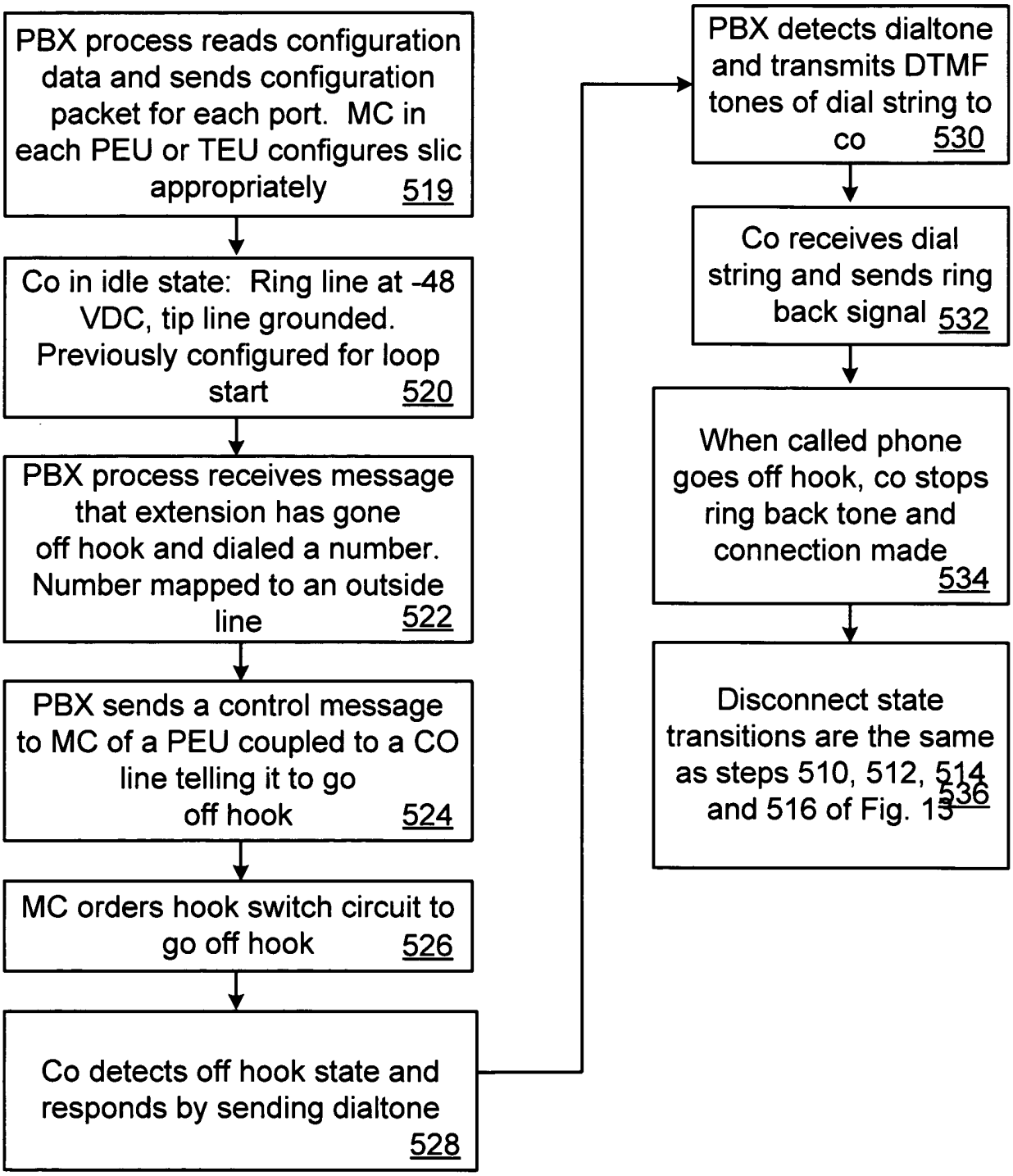


FIG. 14

Ground Start Signaling Protocol - Analog Co Line - Co Originates Call

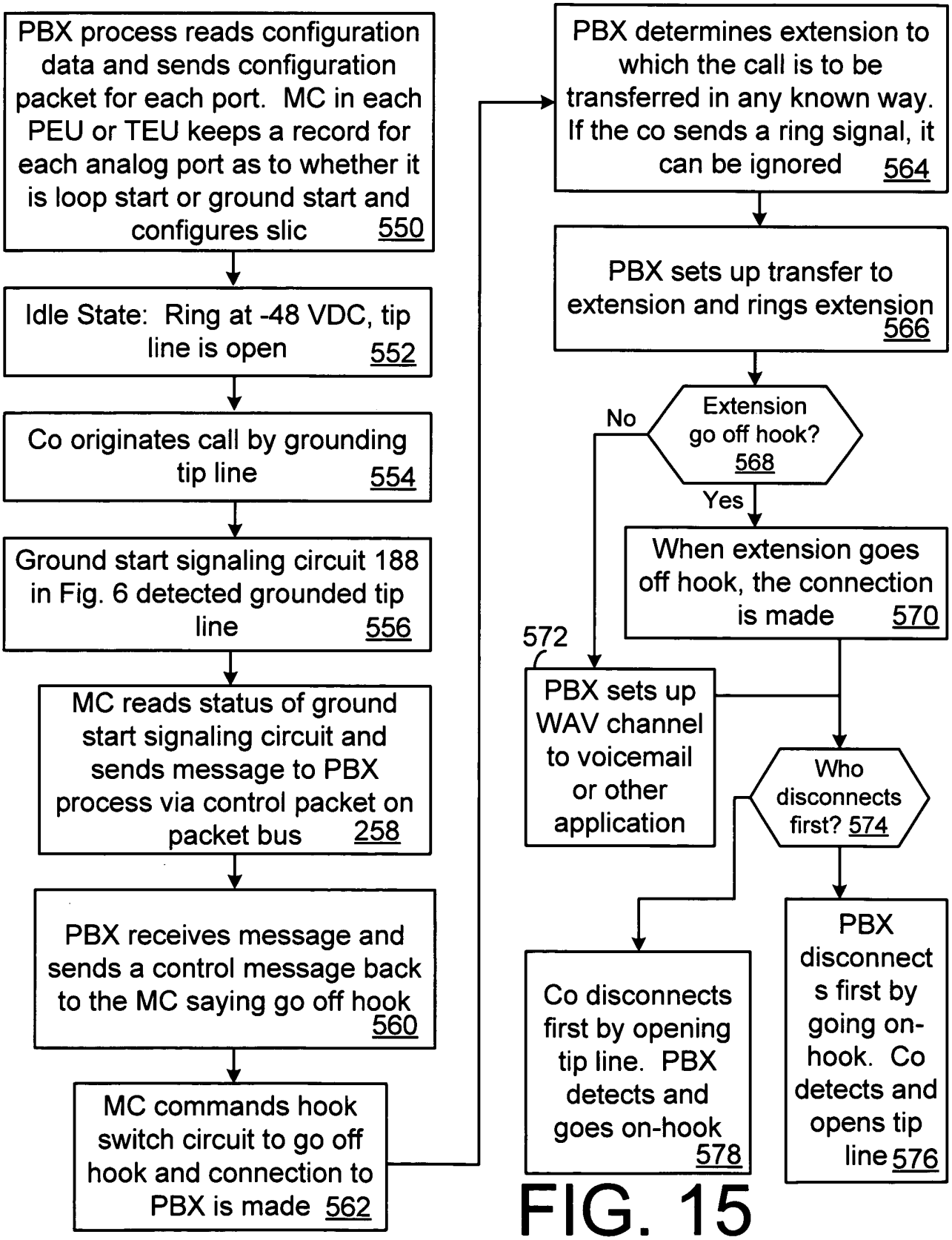


FIG. 15

Ground Start Signaling Protocol - Analog Co Line - PBX Originates Call

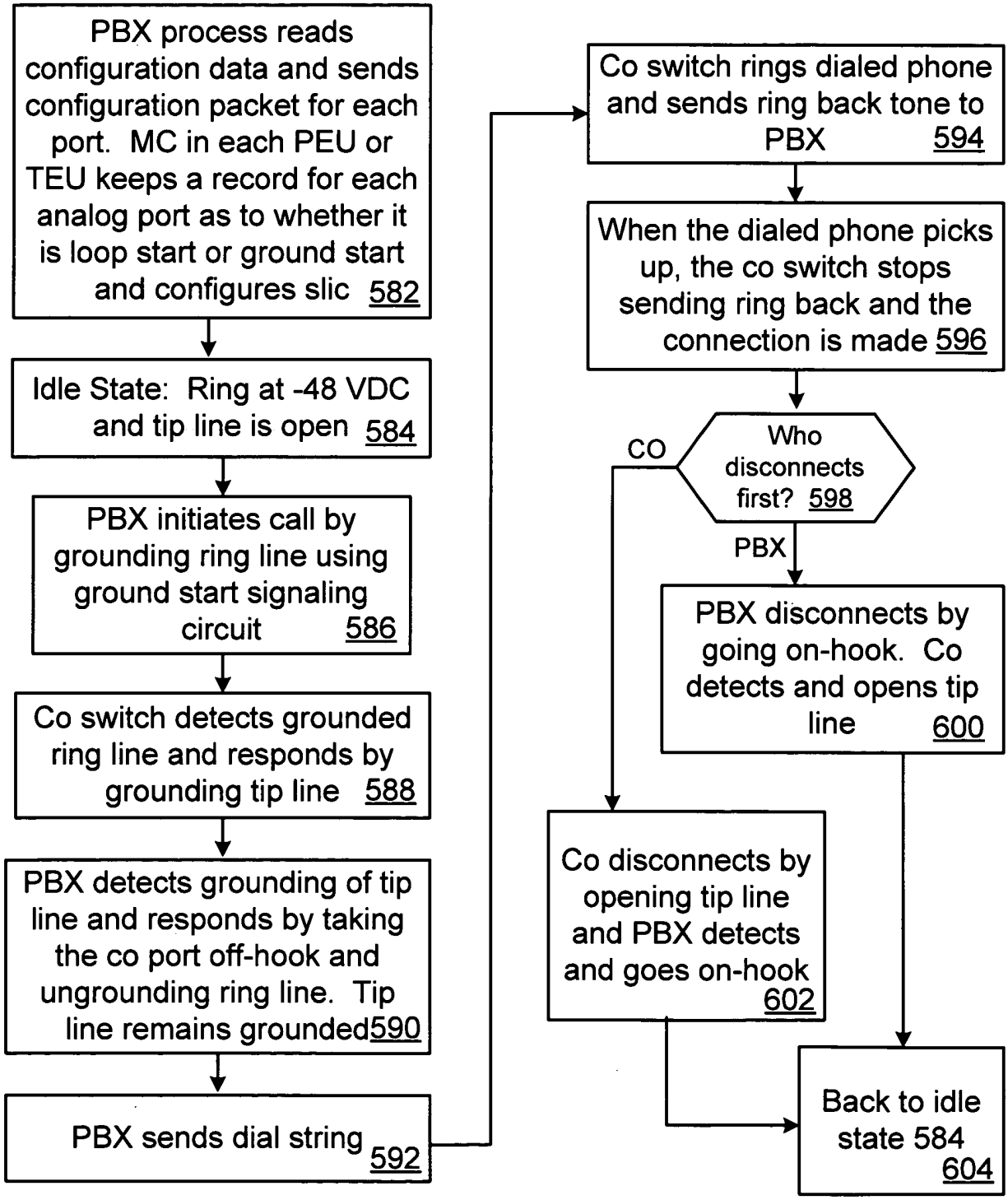


FIG. 16

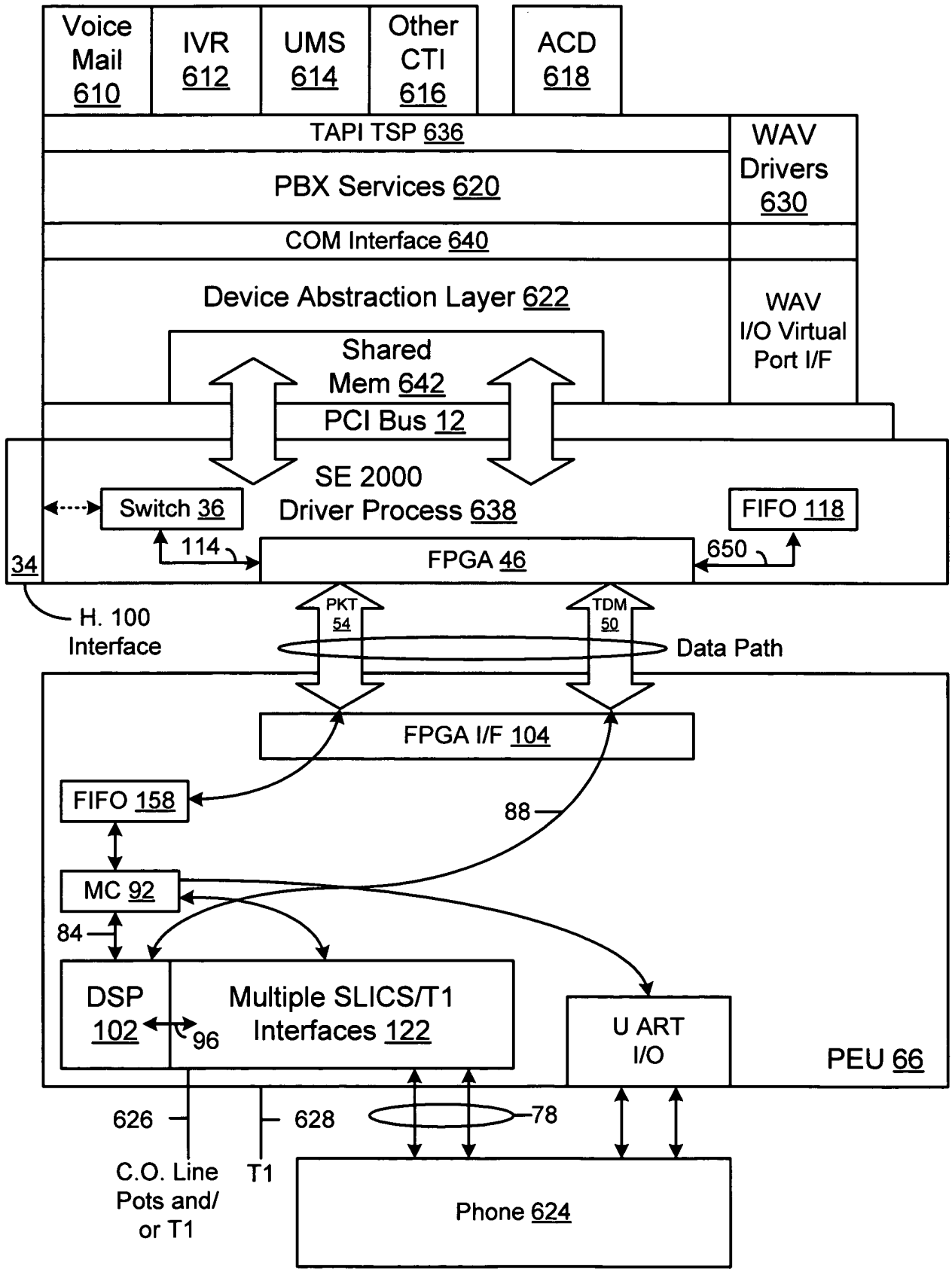
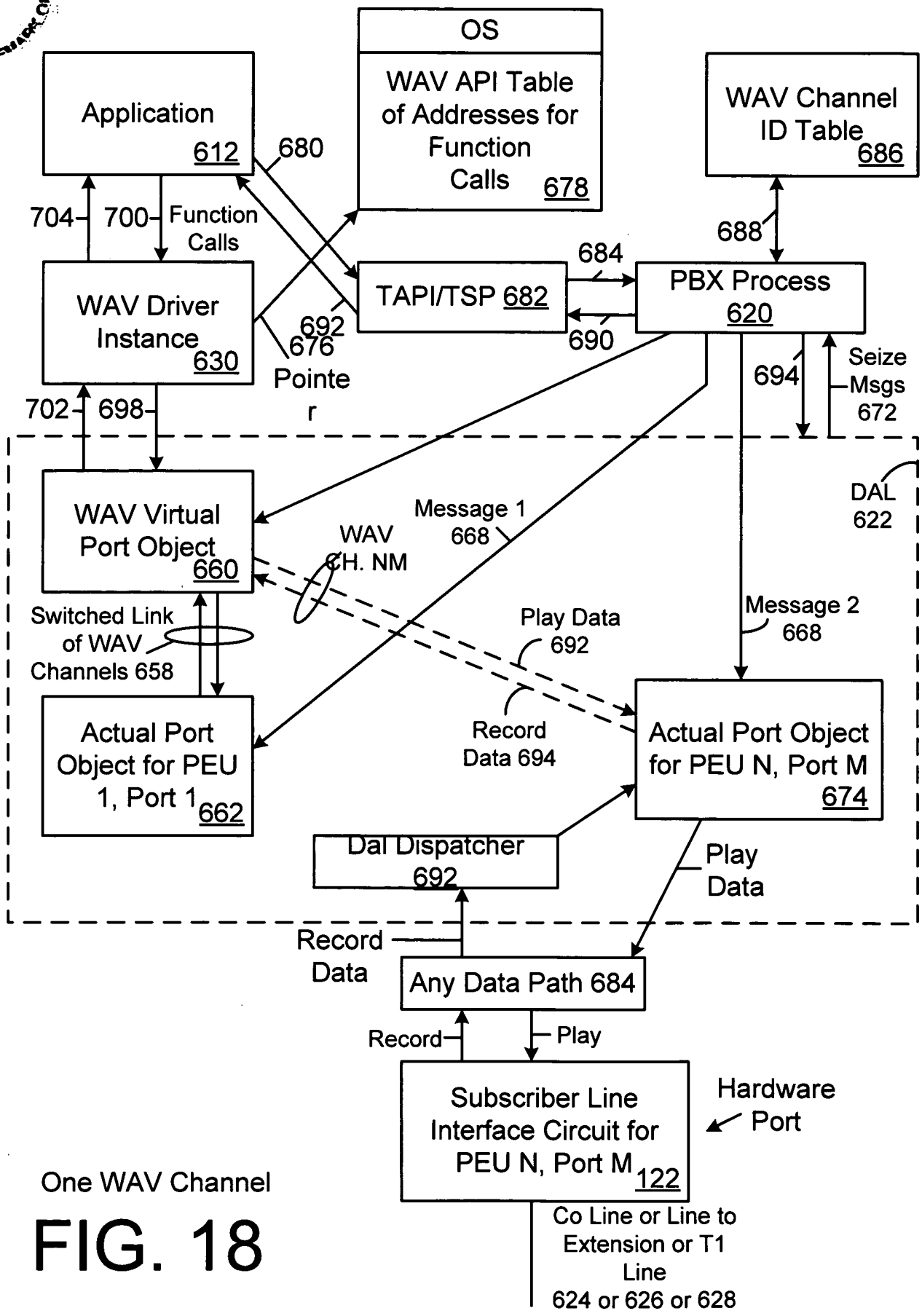


FIG. 17



One WAV Channel

FIG. 18